OTIMIZAÇÃO DO PROBLEMA DE DIMENSIONAMENTO DE LOTES: A INFLUÊNCIA DE PARÂMETROS DO BRANCH-AND-BOUND E DO CORTE DE GOMORY NO DESEMPENHO DA SOLUÇÃO.

PEDRO ROCHAVETZ DE LARA ANDRADE - pedro.rochavetz@gmail.com
UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ - UTFPR - LONDRINA

MATHEUS ARTIOLI LEANDRIN - matheus_leandrin@yahoo.com.br
UNIVERSIDADE ESTADUAL PAULISTA - UNESP - BAURU-FEB

LUIZ HENRIQUE CHERRI - luizcherri@gmail.com
UNIVERSIDADE DE SÃO PAULO - USP - SÃO CARLOS

ADRIANA CRISTINA CHERRI - adriana@fc.unesp.br
UNIVERSIDADE ESTADUAL PAULISTA - UNESP - BAURU-FC

Área: *6 - PESQUISA OPERACIONAL*
Sub-Área: *6.4 - MODELAGEM, ANÁLISE E SIMULAÇÃO*

Resumo: *O PROBLEMA DE DIMENSIONAMENTO DE LOTES É BASTANTE COMUM AO SE ABORDAR O MEIO INDUSTRIAL. PELA NATUREZA COMPLEXA E INTER-RELACIONADA DESTE PROBLEMA PRÁTICO, O TEMPO DE PROCESSAMENTO COMPUTACIONAL DE MÉTODOS EXATOS AUMENTA CONSIDERAVELMENTE QQUANDO SE TRATAM DE PROBLEMAS DE GRANDE PORTE. TRATA-SE DE UM PROBLEMA QUE PODE SER FORMULADO COMO UM PROBLEMA DE PROGRAMAÇÃO LINEAR INTEIRA MISTA, QUE É TRADICIONALMENTE RESOLVIDO ATRAVÉS DE TÉCNICAS COMO O BRANCH-AND-BOUND E ALGORITMOS DE CORTE. VISANDO A MELHORIA DO DESEMPENHO COMPUTACIONAL DOS MÉTODOS DE SOLUÇÃO, NESTE TRABALHO A INFLUÊNCIA DE PARÂMETROS DO MÉTODO BRANCH-AND-BOUND É ANALISADA, BEM COMO DA UTILIZAÇÃO OU NÃO DO ALGORITMO DE CORTE DE GOMORY. COM O TESTE REALIZADO, CONCLUI-SE QUE A UTILIZAÇÃO DO CORTE DE GOMORY DIMINUI O NÚMERO DE NÓS ANALISADOS PELO BRANCH-AND-BOUND, CONTUDO NÃO DIMINUI O TEMPO TOTAL DE PROCESSAMENTO.*

Palavras-chaves: *DIMENSIONAMENTO DE LOTES; PROGRAMAÇÃO LINEAR INTEIRA MISTA; BRANCH-AND-BOUND; CORTE DE GOMORY.*

LOT SIZING PROBLEM OPTIMIZATION: INFLUENCES OF BRANCH-AND-BOUND AND GOMORY CUT PARAMETERS ON THE SOLUTION PERFORMANCE.

Abstract: *THE LOT SIZING PROBLEM IS A USUAL PROBLEM IN THE INDUSTRIAL ENVIRONMENT. DUE TO THE COMPLEX AND INTERRELATED ORIGIN OF THIS PRACTICAL PROBLEM, THE PROCESSING COMPUTATIONAL TIME OF EXACT METHODS CONSIDERABLY INCREASES WHEN LARGE PROBLEMS AREE TREATED. IT IS A PROBLEM THAT CAN BE FORMULATED AS A MIXED INTEGER LINEAR PROGRAMMING PROBLEM, WHICH IS TRADITIONALLY SOLVED BY TECHNIQUES AS BRANCH-AND-BOUND AND CUTTING ALGORITHMS. AIMING THE IMPROVEMENT OF COMPUTATIONAL PERFORMANCE ON SOLUTION METHODS, IN THIS PAPER THE INFLUENCE OF PARAMETERS FROM BRANCH-AND-BOUND IS ANALYZED, SUCH AS THE UTILIZATION OR NOT OF THE GOMORY CUTTING ALGORITHM. WITH THE PERFORMED TEST, IT WAS POSSIBLE TO CONCLUDE THAT THE UTILIZATION OF GOMORY CUTTING ALGORITHM REDUCES THE NUMBER OF TIES ANALYZED BY BRANCH-AND-BOUND TECHNIQUE, HOWEVER IT DOES NOT REDUCE THE TOTAL PROCESSING TIME.*

Keyword: *LOT SIZING; MIXED INTEGER LINEAR PROGRAMMING; BRANCH-AND-BOUND; GOMORY CUTTING.*

1. Introdução

Com o aumento da concorrência industrial, cresce a importância de um bom aproveitamento de todos os recursos envolvidos no processo produtivo. Os diversos métodos de otimização da Pesquisa Operacional (Programação Linear, Programação Inteira, Métodos Heurísticos) são ferramentas que auxiliam a tomada de decisões em diversos ambientes, e que se bem utilizadas, se traduzem em última instância, em um melhor aproveitamento de recursos.

O problema de dimensionamento de lotes é muito comum no meio industrial, pois empresas e setores que trabalham com a fabricação de itens (produtos), comumente podem minimizar os custos envolvidos na utilização de seus recursos (máquinas, equipamentos). Além de estabelecer uma boa relação entre itens produzidos e recursos utilizados, o problema de dimensionamento de lotes também auxilia na resolução de conflitos existentes dentro de um horizonte de planejamento: decisões sobre o que produzir, o quanto produzir e quando produzir. Todavia, o método de solução computacional envolvido em um problema de dimensionamento de lotes deve ser eficiente quanto ao resultado e tempo de processamento.

O tempo de processamento pode tornar-se restrição quando se tratam de problemas reais, de alta complexidade ou de grande porte, inviabilizando a utilização de métodos exatos (Andrade, 2014). Nestes casos, o estudo da alteração de parâmetros dos métodos torna-se fundamental, visando o aumento da eficiência do método e a diminuição do tempo de processamento. Consequentemente, a análise da influência de parâmetros de resolução pode aumentar a chance de viabilidade da utilização de métodos exatos em problemas de alta complexidade e grande porte.

O problema de dimensionamento de lotes foi abordado neste trabalho como um problema de Programação Linear Inteira Mista (PLIM). Para sua resolução, o pacote computacional LINGO® foi utilizado e, sabe-se que esta ferramenta utiliza a técnica *Branch-and-Bound* para a resolução de problemas de Programação Linear Inteira Mista (PLIM). Portanto, foi estudada a influência no desempenho de resolução deste problema quando há alteração de parâmetros, como a estratégia de ramificação, preferenciando a ramificação das variáveis binárias, e a utilização conjunta do algoritmo de Corte de Gomory.

2. Revisão de literatura

2.1. O Problema de Dimensionamento de Lotes

Um dos problemas que envolvem as decisões de planejamento de produção industrial diz respeito ao dimensionamento de lotes. Tal problema começou a ser estudado por Wagner e Whitin (1958), e o primeiro modelo resolvia problemas monoestágio, não capacitados com custo de preparo e único-item.

Para entender o contexto o problema de dimensionamento de lotes, Karimi (2003) aborda alguns tópicos sobre a formulação e complexidade desses problemas. São eles:

- Itens: Quando apenas um produto é produzido, tem-se o sistema tipo único-item, logo, quando itens distintos são produzidos, tem-se o sistema multi-itens.
- Estágios: No sistema multiestágio, um único item é composto por mais de um estágio no setor produtivo e a demanda depende dos estágios anteriores. Desta forma, pode-se dizer que a demanda é dependente. No sistema produtivo monoestágio, o item é produzido em um único estágio de maneira direta. As demandas são satisfeitas pela quantidade de produtos finais, e pode-se dizer que a demanda é independente.
- Demanda: A demanda pode ser dinâmica, quando há variação de demanda no tempo, ou estática, caso contrário. Ela pode ser determinística, se os valores forem conhecidos, ou estocástica, se a demanda seguir uma distribuição de probabilidade.
- Horizonte de Planejamento: Pode ser finito ou infinito, está relacionado ao tempo considerado para o planejamento da produção.

Capacidade: Se existe um limite máximo para a produção de itens, o problema é capacitado. O problema é não capacitado no caso de não existir limite de itens a serem produzidos.

- Tempo e custo de preparo: O tempo de preparo está diretamente relacionado ao custo do item. Este tempo também é chamado de *setup*, e equivale ao tempo necessário para preparação das máquinas antes de iniciar o ciclo produtivo.
- Estoque: Diz respeito à quantidade de itens excedentes armazenados, logicamente, existe um custo para a armazenagem destes itens.
- Atraso na entrega: Uma demanda que não é atendida no período correspondente pode ser entregue no período seguinte, entretanto é atribuído um custo ou uma penalização por este atraso.

O problema de dimensionamento de lotes evoluiu com o passar dos anos. Estudos foram feitos analisando as possíveis variações que o problema pode apresentar. A literatura aborda algumas destas variações: múltiplos itens (Gicquel e Minoux, 2014); processo de produção monoestágio e multiestágio (Almeder *et al*. 2014); considerando períodos com máquinas paralelas (Araujo *et al*. 2015); modelos que propõem soluções a partir algoritmos de seleção (Pawlowski e Szlapka, 2016), e ainda modelos que analisam cruzamento de *setups* entre as mudanças de períodos (Fiorotto *et al*. 2016).

Zhengyang e Guiping (2016) propõem um modelo que não custeia o *setup* entre itens idênticos, permitindo um melhor aproveitamento da capacidade disponível no período em questão, além de reduzir custos provenientes do excesso de *setups*.

Os autores aprimoraram modelos para problemas de dimensionamento de lotes que têm como objetivo minimizar uma função que considera os custos provenientes do processo produtivo, por exemplo, custos de fabricação, estoque, *setup* e custos extras, como penalizações por atraso na entrega. As restrições dos modelos variam de acordo com o problema a ser resolvido, entretanto, em geral a solução destes problemas informam, por exemplo, quais produtos devem ser produzidos, em quais quantidades e em qual período, de forma a atender a demanda dentro da melhor relação de custo-benefício possível.

2.2. Métodos de Resolução

Para a resolução de problemas de PLIM, os métodos mais utilizados são os baseados em planos de corte e na técnica *Branch-and-Bound* (Arenales *et. al*, 2015). O método que combina as duas abordagens é conhecido como *Branch-and-Cut*, e foi desenvolvido por Padberg e Rinaldi (1991). O *Branch-and-Bound* é um método exato de enumeração implícita e, mesmo não avaliando todas as soluções possíveis, garante o alcance da solução ótima (Taha, 2008). Conforme Cury (1999), esta técnica é utilizada para a otimização de problemas com variáveis inteiras, e diminui a região factível do problema eliminando soluções fracionárias através da inserção de novas restrições. Dessa forma, o *Branch-and-Bound* cria uma árvore de nós que representam subproblemas, sendo que cada um deles é resolvido isoladamente, como um problema de Programação Linear, através do método Simplex. Os subproblemas infactíveis bem como os de pior qualidade são eliminados, seguindo o processo até que a solução inteira ótima seja obtida (Cury, 1999).

Existem maneiras de eliminar soluções de baixa qualidade, através da utilização de limitantes. Segundo Kawamura (2006), num problema de minimização o limitante superior é o melhor valor conhecido e viável da função objetivo, e soluções com valor superior a este limitante são descartadas sem serem ramificadas. Além disso, diferentes estratégias de ramificação podem ser utilizadas no desenvolvimento do *Branch-and-Bound*. Três estratégias bastante difundidas são a busca em profundidade, a busca em largura e a busca pelo nó de maior limitante superior. A busca em profundidade se baseia em explorar primeiro os últimos nós encontrados, a busca em largura explora primeiro os primeiros nós encontrados e, por fim, a busca pelo nó de maior limitante superior explora sempre o nó com o melhor valor da função objetivo (Arenales *et. al*, 2015).

A resolução de problemas de Programação Inteira, através do *Branch-and-Bound* é amplamente estudada na literatura, desde o seu desenvolvimento, por Land e Doig (1960). Contudo, a influência dos seus parâmetros na eficiência do método foi estudada por poucos autores. Entre eles, Kawamura (2006) propôs uma estratégia para o *Branch-and-Bound* que integra a do nó mais promissor com a estratégia em profundidade, aplicada à programação de tarefas de uma máquina com datas de entrega comum. Segundo o autor, a seleção do nó a ser ramificado pode influenciar muito no tempo de processamento da técnica.

A tese de Pereira (2007) propõe uma regra de ramificação para o *Branch-and-Bound* aplicado ao problema de p-medianas. A estratégia se baseia na divisão dos subproblemas em dois grupos, identificando pares de vértices com características similares. Como resultado, Pereira (2007) afirma que foi possível reduzir a quantidade média de nós analisados pela técnica, e assim, o tempo computacional demandado.

Scarpin *et. al* (2008) estudaram o fluxo de pacientes que buscam atendimento médico no estado do Paraná como um problema de p-medianas. Os autores utilizam uma técnica que prioriza a ramificação de variáveis levando em conta se elas representam pontos que são atendidos por uma mesma mediana ou não. Este método também elimina soluções não promissoras antes da sua ramificação, economizando tempo de processamento. Com as técnicas utilizadas, o tempo computacional foi satisfatório, sendo sua principal influência a relação entre o número de pontos e o número de medianas do problema.

Oliveira e Morabito (2006) utilizaram técnicas baseadas no *Branch-and-Bound* para otimizar o carregamento de paletes. Na aplicação proposta, a escolha da variável a ser ramificada foi feita através da seleção do melhor desempenho entre três estratégias. Na primeira escolhe-se a variável que representa uma alocação de caixas sem espaços no palete; na segunda escolhe-se a variável com maior valor Lagrangiano e, na terceira escolhe-se a variável que mais aparece nas soluções analisadas. Além disso, os autores estudaram a necessidade de se fazer ou não um número maior de iterações para encontrar o nó inicial do problema e a viabilidade de se gerar os limitantes inferiores iniciais através de relaxação Lagrangiana e *Surrogate*. Os autores afirmam que a primeira estratégia de escolha da variável a ser ramificada gera um melhor resultado. Além disso, não é necessário realizar mais iterações na geração da solução inicial, contudo, a utilização da técnica para gerar limitantes inferiores melhora o desempenho do método proposto.

3. Problema de Dimensionamento de Lotes

O problema resolvido neste trabalho utilizou como referência os modelos propostos por Robinson *et al.* (2009) e Jans e Degraeve (2007) para problemas capacitados de dimensionamento de lotes com demanda determinística.

As principais características do problema, quanto à sua formulação, levaram em consideração um único recurso para processar todos os itens, a diversidade destes itens, a restrição de capacidade do recurso, múltiplos períodos disponíveis para atender uma demanda estática e determinística para cada item. Todos os custos de *setups* e produção de cada item são conhecidos, cuja soma deve ser minimizada.

O modelo proposto tem como objetivo dimensionar quais itens e em quais quantidades devem ser produzidos nos respectivos períodos, de forma a atender toda a demanda ao menor custo possível. Dessa forma, os índices necessários são:

- t: 1, 2, ..., T : períodos de tempo;
- i :1, 2, ..., N : itens;

Parâmetros:

- d_i: demanda total de cada item i;
- c_i: custo unitário de produção do item i;
- b_i: tempo unitário de produção da item i;
- s_i: custo unitário de *setup* para a produção do item i;
- sp_i: tempo unitário de *setup* para a produção do item i;
- Cap_t: capacidade disponível no período t;
- L_{it} limitante que indica o mínimo entre a capacidade restante disponível se o item i é produzido no período t, e a demanda acumulada do período t ao período T.

$$L_{it} = \min\{\frac{Cap_t - sp_i}{b_i}, \sum_{\tau=1}^{T} di\tau\} \qquad i = 1, \dots, N, t = 1, \dots, T$$

Variáveis:

- x_{it}: quantidade produzida do item i no período t;
- y_{it}: variável binária que indica a produção ou não da peça i no período t;

Considerando os índices e parâmetros apresentados, o modelo proposto é o seguinte:

$$min \sum_{i=1}^{N}\sum_{t=1}^{T}(c_i\, x_{it} + s_i\, y_{it}) \quad (1)$$

Sujeito à:

$$\sum_{i=1}^{N}(b_i\, x_{it} + sp_i\, y_{it}) \leq Cap_t \qquad t = 1, \dots, T \quad (2)$$

$$\sum_{i=1}^{N} x_{it} = d_i \qquad t = 1, \dots, T \quad (3)$$

$$x_{it} \leq L_{it} y_{it} \qquad i = 1, \dots, N, t = 1, \dots, T \quad (4)$$

$$x_{it} \in \mathbb{Z}_+ , y_{it} \in \{0,1\} \qquad i = 1, \dots, N, t = 1, \dots, T \quad (5)$$

A função objetivo (1) minimiza o custo total de produção e *setup*. A restrição (2) garante que o tempo de produção de cada item somado ao tempo de *setup* não ultrapassem o tempo disponível para a máquina no período t. A restrição (3) garante que a soma das quantidades dos itens i produzidos em todos os períodos t deve ser igual à demanda de cada tipo de item. A restrição (4) garante que haverá produção do item i no período t somente se houver preparação de *setup* ($y_{it} = 1$). A restrição (4) garante uma quantidade mínima de itens produzidos no período e, por fim, a restrição (5) indica o domínio das variáveis.

Para resolver o modelo de otimização do problema de dimensionamento de lotes, bem como analisar o desempenho do método *Branch-and-Bound* aliado às técnicas de corte, um problema fictício foi elaborado, sendo apresentado a seguir juntamente com a metodologia de resolução.

4. Testes computacionais

Uma empresa utiliza um equipamento para processar 7 itens. A disponibilidade deste equipamento está dividida em 5 períodos com capacidade de 360 minutos cada. As características relevantes de cada item são: custo de produção (c_i) em Reais; custo de *setup* (s_i) em Reais; demanda final (d_i) em unidades; tempo de produção (b_i) em minutos; e tempo de *setup* (sp_i) em minutos. O problema consiste em determinar a quantidade em que cada item deve ser produzido em cada período de forma a atender toda a demanda, respeitar a capacidade de produção de cada período e minimizar o custo total de produção.

Dados do problema:

Período t = (1; 2; 3; 4; 5)

Itens i = (A; B; C; D; E; F; G)

Custo de produção c_i = (60; 65; 90; 100; 80; 75; 50)

Custo de *setup* s_i = (14; 14; 18; 24; 16; 20; 28)

Demanda total d_i = (33; 25; 45; 28; 20; 32; 42)

Tempo de produção b_i = (67; 9; 10; 8; 8; 5)

Tempo de *setup* sp_i = (7; 7; 9; 12; 8; 10; 14)

Para resolver o problema, a linguagem algébrica de programação LINGO® foi utilizada. O Software LINGO® utiliza, para problemas com funções quadráticas e restrições lineares, um procedimento heurístico de Busca Local com estratégia *Multistart Solver*. Já quando se trata da resolução de problemas de Programação Linear e Programação Inteira, o sistema utiliza as técnicas Simplex e *Branch-and-Bound* (LINDO, 2017).

O problema proposto foi modelado e resolvido pelo software LINGO® 13.0. Foram analisadas as influências da alteração de parâmetros de resolução como: utilização conjunta de diferentes algoritmos de cortes; alteração da estratégia de ramificação do *Branch-and-Bound*. De acordo com Kawamura (2006), o tempo de convergência do *Branch-and-Bound* está fortemente relacionado à variável escolhida para ser ramificada, sendo que existem variáveis que influenciam mais na solução do que outras. O LINGO® permite definir uma escala numérica de preferência entre as variáveis como um artifício que pode melhorar a convergência do método. Além disso, esta ferramenta permite que, para problemas com variáveis binárias, estas variáveis sejam ramificadas antes de outras variáveis inteiras.

Outra possibilidade de alteração na resolução do método *Branch-and-Bound* é a escolha de um ou mais métodos de corte para serem aplicados de forma conjunta, aumentando assim a sua eficiência. Neste trabalho, será estudada a influência da utilização ou não do algoritmo de Corte de Gomory.

Dessa forma, o estudo aqui realizado apresenta a resolução de um problema de dimensionamento de lotes fictício, utilizando diferentes parâmetros de resolução do método *Branch-and-Bound*. São analisadas as diferenças no tempo de processamento, no número de iterações e no número de nós analisados quando se preferência ou não a ramificação das variáveis binárias e quando se utiliza ou não o algoritmo de Corte de Gomory.

5. Resultados

O problema foi resolvido através do *software* LINGO® 13.0, executado em um computador com processador *Intel Core* i5, 4 GB de memória RAM e sistema operacional Windows® de 64 bits. A solução encontrada, em relação à quantidade de cada item a se produzir em cada período, para cada uma das formas de resolução analisadas é mostrada a seguir.

Utilizando a preferência de ramificação de variáveis binárias, sem o Corte de Gomory:

- Período t_1 = (x_{B1} = 25; x_{E1} = 20)
- Período t_2 = (x_{A2} = 12; x_{F2} = 32)
- Período t_3 = (x_{C3} = 6; x_{D3} = 28)
- Período t_4 = (x_{A4} = 21; x_{G4} = 42)
- Período t_5 = (x_{C5} = 39)

Utilizando a preferência de ramificação de variáveis binárias, com o Corte de Gomory:

- Período t_1 = (x_{C1} = 39)
- Período t_2 = (x_{A2} = 19; x_{G2} = 42)
- Período t_3 = (x_{A3} = 14; x_{F3} = 32)
- Período t_4 = (x_{B4} = 25; x_{E4} = 20)
- Período t_5 = (x_{C5} = 6; x_{D5} = 28)

Sem a preferência de ramificação de variáveis binárias, sem o Corte de Gomory:

- Período t_1 = (x_{A1} = 19; x_{G1} = 42)
- Período t_2 = (x_{C2} = 39)
- Período t_3 = (x_{C3} = 6; x_{D3} = 28)
- Período t_4 = (x_{A4} = 14; x_{F4} = 32)
- Período t_5 = (x_{B5} = 25; x_{E5} = 20)

Sem a preferência de ramificação de variáveis binárias, com o Corte de Gomory:

- Período t_1 = (x_{C1} = 39)
- Período t_2 = (x_{B2} = 25; x_{E2} = 20)
- Período t_3 = (x_{A3} = 12; x_{F3} = 32)
- Período t_4 = (x_{C4} = 6; x_{D4} = 28)
- Período t_5 = (x_{A5} = 21; x_{G5} = 42)

O valor da função objetivo (custo de produção) para todos os casos é de 16.721, visto que este é o valor ótimo para este problema. O diferencial analisado é referente ao tempo de processamento, número de iterações e número de nós analisados em cada caso. A Tabela 1 ilustra o tempo computacional demandado em cada forma de resolução.

TABELA 1 – Tempo de processamento (em segundos).

		Corte de Gomory	
Preferência de ramificação Var. Binárias		Sim	Não
	Sim	49	66
	Não	61	60

Fonte: Elaborado pelos autores (2017).

Nota-se pela Tabela 1 que a utilização do Corte de Gomory aliado à preferência de ramificação de variáveis binárias melhorou significativamente o desempenho computacional para este caso. Para se concluir que o Corte de Gomory reduz o tempo de processamento, mais casos devem ser analisados. De qualquer forma, neste caso a média do tempo de processamento foi significativamente menor utilizando este algoritmo (55 segundos), do que quando ele não foi utilizado (63 segundos). A Tabela 2 ilustra o número de iterações de cada caso:

TABELA 2 – Número de iterações.

		Corte de Gomory	
Preferência de ramificação Var. Binárias		Sim	Não
	Sim	256.942	305.477
	Não	282.631	275.110

Fonte: Elaborado pelos autores (2017).

Analisando a Tabela 2, pode-se perceber que o número de iterações está diretamente relacionado ao tempo de processamento. A preferência pela ramificação de variáveis binárias não demonstra melhorar ou piorar significativamente o desempenho da ferramenta. A Tabela 3 ilustra o número de nós analisados em cada forma de resolução.

TABELA 3 – Número de nós analisados.

		Corte de Gomory	
Preferência de ramificação Var. Binárias		Sim	Não
	Sim	17.473	21.259
	Não	18.020	20.031

Fonte: Elaborado pelos autores (2017).

Pela Tabela 3 fica claro que a utilização do Corte de Gomory, neste caso, reduziu significativamente o número de nós a serem analisados na resolução do *Branch-and-Bound*. O LINGO® permite que a decisão em relação à qual variável ramificar seja tomada pela própria ferramenta. Para efeito de comparação, o problema foi resolvido desta forma, selecionando também a utilização conjunta de todas 12 as técnicas de corte disponibilizadas. A solução ótima encontrada foi a seguinte:

- Período t_1 = (x_{A1} = 12; x_{F1} = 32)
- Período t_2 = (x_{B2} = 25; x_{E2} = 20)
- Período t_3 = (x_{C3} = 6; x_{D3} = 28)
- Período t_4 = (x_{C4} = 39)
- Período t_5 = (x_{A5} = 21; x_{G5} = 42)

O tempo de processamento foi de 65 segundos, foram realizadas 316.485 iterações e o número de nós analisados foi de 11.934, ou seja, o número de nós analisados foi significativamente menor do que todos os outros casos, mostrando que a utilização de diversas técnicas de corte auxilia na redução da necessidade de análise de nós pelo método. Contudo, estas técnicas utilizadas também demandam tempo adicional de processamento, por isso, não houve redução no tempo de processamento nesta forma de resolução.

6. Considerações Finais

Neste trabalho, o problema de dimensionamento de lotes foi resolvido utilizando o *software* LINGO® e analisando a influência de parâmetros do método *Branch-and-Bound*, bem como a utilização ou não do algoritmo de Corte de Gomory. Com os testes realizados, foi possível perceber que a utilização do algoritmo de Corte de Gomory reduz a quantidade de nós a serem analisados pelo método e, além disso, a utilização de diversas técnicas de corte reduz ainda mais a necessidade de análise de nós. Contudo, estas técnicas não reduzem o número total de iterações nem o tempo de processamento. O mesmo se pode afirmar em relação à preferência de ramificação de variáveis binárias, que não demonstrou melhoria de desempenho para o problema proposto.

Para trabalhos futuros, sugere-se que problemas de maior porte sejam estudados, em que as diferenças de desempenho dos diferentes métodos possam ser mais ressaltadas. Além disso, seria importante para a validação dos resultados, que o estudo da influência destes parâmetros fosse realizado em relação a outros tipos de problemas. Por fim, sugere-se que em trabalhos futuros, outros parâmetros do *Branch-and-Bound* sejam analisados, como estratégias de ramificação e seleção de variáveis.

Referências

ALMEDER, C. KLABJAN, D. TRAXLER, R. LOBO, A. B. Lead time considerations for the multi-level capacitated lot-sizing problem. *European Journal of Operational Research*. vol. 241, n. 3, p. 727-738, 2014.

ANDRADE, P. R. L. *Otimização na geração de grade horária escolar através de um modelo matemático e das meta-heurísticas busca local e iterated local search*. Dissertação (Mestrado). Programa de Pós-Graduação em Engenharia de Produção - Universidade Federal do Paraná. Curitba, 2014.

ARENALES, M. ARMENTANO, V. MORABITO, R. YANASSE, H. *Pesquisa Operacional.* 2ª ed. Rio de Janeiro: Elsevier, 2015.

CURY, R. M. *Uma abordagem difusa para o problema de flow-shop scheduling.* p. 88. Tese (Doutorado) – Programa de Pós-Graduação em Engenharia de Produção, Universidade Federal de Santa Catarina, Florianópolis, 1999.

FIOROTTO, D. J. ARAUJO, S. A. JANS, R. Hybrid methods for lot sizing on parallel machines. *Computer and Operations Research*. vol. 63, p. 136-148, 2015.

FIOROTTO, D. J. ARAUJO, S. A. JANS, R. An analysis of formulations for the capacitated lot sizing problem with setup crossover. *Computers and Industrial Engineering*. vol. 106, p. 338-350, 2016.

GICQUEL, C. MINOUX, M. Multi-product valid inequalities for the discrete lot-sizing and scheduling problem. *Computer and Operational Research*. vol. 54, p. 12-20, 2014.

JANS, R. DEGRAEVE, Z. Meta-Heuristics for Dynamic Lot Sizing: A Review and Comparison of Solution Approaches. *European Journal of Operational Research,* vol. 177, n. 3, p. 1855-1875, 2007.

KARIMI, B. GHOMI, S. F. WILSON, J. The capacitated lot sizing problem: A review of models and algorithms. *Omega*, vol. 31, p. 365-378, 2003.

KAWAMURA, M. S. *Aplicação do método Branch-and-Bound na programação de tarefas em uma única máquina com data de entrega comum sob penalidades de adiantamento e atraso*. p. 71. Dissertação (Mestrado) – Departamento de Engenharia de Produção, Escola Politécnica da Universidade de São Paulo, São Paulo, 2006.

LAND, A. H. DOIG, A. G. An Automatic Method of Solving Discrete Programming Problems. *Econometrica: Journal of the Econometric Society*, vol. 28, n. 3, p. 497-520, 1960.

LINDO. Lindo Systems Inc. Disponível em: <http://www.lindo.com/doc/online_help/lingo15_0/multi-start_attempts.htm>. Acesso em 29/04/2016.

OLIVEIRA, L. K. MORABITO, R. Métodos exatos baseados em relaxações lagrangeanas e surrogate para o problema de carregamento de paletes do produtor. *Pesquisa Operacional*, vol. 26, n. 2, p. 403-432, 2006.

PADBERG, M. RINALDI, G. A branch-and-cut algorithm for the resolution of large-scale symmetric traveling salesman problems. *Society for Industrial and Applied Mathematics*. vol. 33, n. 1, p. 60-100, 1991.

PAWLOWSKI, G. SZLAPKA, J. O. Scheduling and lot size problems for variable range of products using GA-based method. *International Federation of Automatic Control*. vol. 49, n. 12, p. 662–667, 2016.

PEREIRA, M. A. *Um método Branch-and-Price para problemas de localização de p-medianas.* Tese (Doutorado) – Programa de Pós-Graduação em Computação Aplicada, Instituto Nacional de Pesquisas Espaciais, São José dos Campos, 2007.

ROBINSON, P. NARAYANAN, A. SAHIN F. Coordinated Deterministic Dynamic Demand Lot-Sizing Problem: A Review of Models and Algorithms. *Omega*. vol. 37, n. 1, p. 3-15, 2009.

SCARPIN, C. T. STEINER, M. T. A. DIAS, G. J. C. STEINER NETO, P. J. Otimização no serviço de saúde no estado do Paraná: fluxo de pacientes e novas configurações hierárquicas. *Gestão e Produção*. vol. 15, n. 2, p. 275-290, São Carlos, 2008.

TAHA, H A. Pesquisa Operacional: uma visão geral. 8ª ed. São Paulo: Pearson Prentice Hall. ISBN 978-85-7605-150-3, 2008.

WAGNER, H. WHITIN, T. Dynamic version of the economic lot size model. *Management Science*, vol. 5, p. 89-96, 1958.

ZHENGYANG, H. GUIPING, H. A two-stage stochastic programming model for lot-sizing and scheduling under uncertainty. *Int. J. Production Economics*. vol. 180, p. 198–207, 2006.